Prius: Hybrid Edge Cloud and Client Adaptation for HTTP Adaptive Streaming in Cellular Networks

Zhisheng Yan, *Student Member, IEEE,* Jingteng Xue, and Chang Wen Chen, *Fellow, IEEE*

***Abstract*—In this paper, we present Prius, a hybrid edge cloud and client adaptation framework for HTTP adaptive streaming (HAS) by taking advantage of the new capabilities empowered by recent advances in edge cloud computing. In particular, emerging edge clouds are capable of accessing application-layer and radio access networks (RAN) information in real time. Coupled with powerful computation support, an edge cloud assisted strategy is expected to significantly enrich mobile services. Meanwhile, although HAS has established itself as the dominant technology for video streaming, one key challenge for adapting HAS to mobile cellular networks is in overcoming the inaccurate bandwidth estimation and unfair bitrate adaptation under the highly dynamic cellular links. Edge cloud assisted HAS presents a new opportunity to resolve these issues and achieve systematic enhancement of quality of experience (QoE) and QoE fairness in cellular networks.**

To explore this new opportunity, Prius overlays a layer of adaptation intelligence at the edge cloud to finalize the adaptation decisions while considering the initial bandwidth-irrelevant bitrate selection at the clients. Prius is able to exploit RAN channel status, client device characteristics as well as application-layer information in order to jointly adapt the bitrate of multiple clients. Prius also adopts a QoE continuum model to track the cumulative viewing experience and an exponential smoothing estimation to accurately estimate future channel under different moving patterns. Extensive trace-driven simulation results show that Prius with hybrid edge cloud and client adaptation is promising under both slow and fast moving environment. Furthermore, Prius adaptation algorithm achieves a near-optimal performance that outperforms exiting strategies.

***Keywords*—Edge cloud, fairness, HTTP adaptive streaming, mobile cellular networks, 3G, 4G, LTE, QoE.**

I. Introduction

THANKS to the proliferation of mobile services and the advancement of cloud computing technologies, *mobile edge computing* (MEC) [1] is becoming an emerging paradigm that offers enhanced user experience to mobile users. Many organizations, e.g., European Telecommunications Standards Institute and 3GPP [2], have been working on the standardization of this new technology. MEC pushes cloud computing capabilities to the edge of mobile cellular networks by deploying high-performance servers within the radio access networks (RAN), e.g., at the base station or radio network controller. In addition to computation and storage support, such *edge cloud* assisted computing is uniquely characterized by real-time access to both application and RAN information. Thus mobile users' experience could be significantly enriched via both service and network adaptation in the edge cloud.

At the same time, video streaming over cellular networks has become one of the most prevalent mobile service. According to a recent study, video traffic will account for nearly three-forths of total mobile data traffic by the end of 2019 [3]. Due to its inherent scalability and versatility, HTTP adaptive streaming (HAS) [4] has been widely recognized as the dominant technology for mobile video delivery. The video source is pre-encoded in several bitrate versions and each version is split into small segments. The client adaptively requests the video segment at each switching point based on per-segment throughput measurement and estimation. That way, the user is expected to receive the most proper video version and achieve satisfactory quality of experience (QoE).

Despite this broad consensus about HAS, we observe that the understanding of rate adaptation strategies for mobile videos is still limited. This might surprise many experts since HAS has been actively studied in multimedia and networking communities. The reason behind this observation is that edge cloud introduces new effects on HAS over cellular networks while little is known regarding the optimal adaptation under this new context.

First, standard throughput based adaptations [5], [6] cannot capture the bandwidth variations in cellular networks. It has been shown that they largely over/underestimate the bandwidth share unless the downloading of a segment can saturate the end-to-end bandwidth[1] [7], [8]. This is attributed to the mismatch between the per-segment throughput and the real bandwidth share. The strong dynamics of cellular bottleneck makes this issue more tricky because channel condition can suddenly degrade just when the high-layer estimation/smoothing/probing approximates the bandwidth share. Thus the instable and unfair playback will be inevitably introduced. More unfortunately, measurement studies [9] have shown that the prevalent assumption of TCP fairness may not be true in cellular networks. This indicates that multiple clients would not achieve fair performance even though they somehow learn the channel information and request the true bandwidth share. Finally, since the multi-client playback on top of TCP bandwidth share may not be fair over cellular networks, it is desired to proactively adjust bitrate of each client. For example, requesting a bitrate lower than the extremely high bandwidth of a client may be more fair than pushing its bitrate to the unfair bandwidth share.

Zhisheng Yan and Chang Wen Chen are with Computer Science and Engineering Department, State University of New York at Buffalo, Buffalo, NY, 14228 (e-mail: {zyan3, chencw}@buffalo.edu)

Jingteng Xue is with Apple Inc., Cupertino, CA, 95014 (e-mail: jingteng_xue@apple.com)

This research is supported by NSF Grants ECCS-1405594.

[1] In this paper, the terms "bandwidth", "bandwidth share" and "available bandwidth" are used interchangeably.

This is because it saves some channel resources for other clients with lower bandwidth share, which will contribute to the overall QoE fairness.

We observe that the fundamental reason behind the aforementioned issues in cellular networks is that the clients are oblivious to the bottleneck radio channel and cannot coordinate with each other to obtain a fair bitrate under the unfair bandwidth share. The emerging edge cloud presents a new opportunity to remedy these issues. As the low-layer RAN information (e.g., instant channel state) is available in edge cloud, the variation of link capacity can be predicted and utilized towards rate adaptation. Besides, edge cloud is a centralized entity that sits within the RAN, e.g., at the base station, and thereby can perform cell-wide joint adaptation. Moreover, thanks to the computational support of edge cloud, sophisticated QoE models and optimal rate adaptation for multiple clients can be developed to maintain the fair perception of users. Therefore, the goal of this research is to study HAS with edge cloud support in order to maximally enhance users' QoE and QoE fairness for a multi-client mobile cell.

One key challenge of the proposed research is to design a edge cloud assisted adaptation framework that is compatible with the client-driven MPEG-DASH standard [4]. We first thoroughly discuss the problems of current infrastructure, confirm the issues via trace-driven simulations, and systematically address the needs of a new adaptation architecture. Then we present *Prius*, an *hybrid edge cloud and client rate adaptation framework* for a single cell with multiple clients. Specifically, Prius overlays a layer of adaptation intelligence at the edge cloud to finalize the adaptation decisions while considering the initial bandwidth-irrelevant bitrate selection at the clients. This allows us to exploit channel information, device characteristics, and application-layer information to jointly optimize the rate adaptation of multiple clients in cellular networks.

Within Prius, we model QoE as *QoE continuum*, where playback quality, interruption, and smoothness are considered, in order to accurately capture the cumulative viewing experience of users. Furthermore, Prius adopts an exponential smoothing based scheme to maximally improve the future channel estimation under different moving patterns. Based on these designs, we formulate a rate adaptation problem at the edge cloud that can guarantee both QoE continuum and QoE fairness among multiple clients while satisfying channel resources and client-side constraint. Accordingly, we propose a heuristic algorithm to efficiently solve the optimization problem.

We validate the designs and algorithms of Prius through extensive trace-driven simulation. The results show that the hybrid adaptation architecture of Prius is promising under both slow and fast moving environment. It demonstrates satisfactory QoE performance in terms of smoothness and fairness. Besides, the proposed rate adaptation algorithm achieves a near-optimal performance that outperforms exiting adaptation algorithms.

To summarize, the contributions of Prius include:

- A *hybrid edge cloud and client* HTTP adaptive streaming framework for mobile cellular networks serving multiple clients (Section III-V).
- A rate adaptation strategy and the associated algorithms that can guarantee both QoE continuum and fairness (Section VI-VII).
- A demonstration of the effectiveness of the proposed edge cloud assisted HAS designs via systematic simulations. (Section VIII).

II. Related Work

Single-client adaptation. Xiang *et al.* proposed a Markov Decision Process based framework, where buffer level and playback variation were considered to ensure the playback quality [10]. Liu *et al.* used a step-wise increase and aggressive decrease adaptation algorithm to quickly match the end-to-end throughput [6]. The authors in [5] validated the effectiveness of MEPG-DASH client logic over commercial softwares under vehicular environment. However, all these algorithms target the single-user and client-side adaptation. They cannot be directly applied to multi-client cellular networks since they are insensitive to QoE fairness. FESTIVE [11] presented the first algorithm to address HAS fairness by harmonic bandwidth estimator, delayed bitrate update and randomized request scheduler. Li *et al.* designed an approach that probes the fair TCP bandwidth share and adapts the bitrate accordingly [8]. Although these two schemes work satisfactorily in stable wired networks thanks to TCP fairness, they may not guarantee fair playback under dynamic cellular links with unfair underlying bandwidth share.

In this paper, we confirm the issues of conventional client-side adaptation. Our Prius system adopts a hybrid edge cloud and client adaptation architecture, where we use the channel information available in edge cloud to accurately manage bandwidth variation and satisfy the device constraint in the client to achieve multi-client optimization and fairness.

Multi-client Adaptation. In [12], the authors stabilized the source traffic from the server for competing clients when oscillations are detected. Another traffic shaping mechanism, wherein the client who enjoyed a higher quality would be firstly downgraded, was developed to improve fairness [13]. In [14], the authors employed feedback control theory to execute the measurement and control at the server side to improve the video playback of multiple clients. These server-side schemes demonstrate the principle of joint adaptation. However, they are not specifically designed for dynamic cellular networks. These schemes impose no constraint on the radio resources, which might still obtain either underestimated or overestimated bitrate for adaptation.

Little work has been done to design HAS over multi-client cellular networks. Some schemes [15]–[17] combined the designs of rate adaptation and resource allocation in order to allow channel-aware playback. They depend on the customized low-layer cellular scheduler, which needs to modify the standard cellular infrastructure with proportional fair scheduler. Others [18], [19] aimed at optimizing the utility function of cellular users. Although the system performance is improved, the client-side decisions are completely overwritten, which could cause issues when client device has certain limitations. Furthermore, the utility function lacks desired connection with user experience. Therefore, there is a significant potential to

move up the design space to the hybrid edge cloud and client adaptation to enhance the QoE of multiple mobile users.

The objective of Prius is fundamentally different from existing works in that Prius probes into the information space of edge cloud and intends to employ a new hybrid adaptation architecture. Prius jointly adapts the bitrate by exploiting the cell-wide channel status, device features, and application information on top of the standardized cellular scheduler. Furthermore, Prius optimizes QoE continuum in order to replace existing instantaneous QoE/QoS functions. Moreover, Prius develops a new channel estimation scheme in order to facilitate all types of users in both fast and slow moving environments.

A QoE continuum driven HAS over multi-client wireless networks was preliminarily developed without the consideration of edge cloud in [20]. In this new research, taking the advantage of edge cloud's recent advances, we have substantially re-designed the framework: (1) an hybrid edge cloud and client side adaptation architecture, (2) an edge cloud assisted optimized rate adaptation strategy, (3) an improved channel estimation, and (4) an enhanced system evaluations.

QoE in HAS. QoE has also been recently included in the design of video streaming systems. For example, bandwidth [21] and power efficiency [22] have been considered as the QoE design objective, respectively. However, no explicit model is provided for QoE measurement. Most QoE models for HAS focused on establishing the relationship between mean opinion score (MOS) and certain encoding/networking parameters at the current viewing moment. For example, QoE was modeled as a function of content type, sender's instant bitrate, current block error rate and mean burst length in [23]. Nevertheless, such schemes were unable to capture the QoE over the entire viewing session. Although some works have attempted to study the importance of temporal factors for QoE, they require human intervention to compute the model [24] or depend on the predefined adaptation logic [25], which makes them difficult to be deployed in general scenarios. In this paper, with the new capacities of edge cloud in real-time access to application layer information, Prius intends to use a QoE continuum model to measure one's cumulative user experience at a particular moment. Prius will apply this QoE measurement scheme into the hybrid edge cloud assisted HAS framework to guide the rate adaptation.

III. Motivations

A. Issues of HAS over Cellular

In this section, we carefully discuss the problems of current HAS over cellular networks.

First, *per-segment throughput cannot capture the bandwidth share*. A typical HAS client measures the throughput for downloading a segment and then requests a bitrate equal to or smaller than the bandwidth share that is predicted from the throughput. Since the requested video bitrate is usually not exactly equal to the available bandwidth, a client will first keep downloading the segment (ON-interval) and then become idle for a while (OFF-interval) before requesting next segment. It has been shown that this ON-OFF pattern will lead to a mismatch between the per-segment throughput and the true

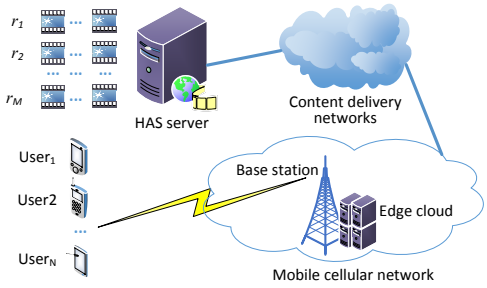


Fig. 1. System Architecture.

bandwidth share [7], [8], which causes over/underestimated segment requests. For example, client A's OFF interval may be overlapped with client B's ON-interval. By dividing the segment size over the length of its ON-interval (less than segment duration), client A will overestimate its bandwidth share because it has no knowledge of client B streaming during its OFF-interval. The bandwidth overestimation will then cause the instable and unfair playback among clients. Note that over/underestimated bandwidth can only be avoided when the segment size saturates the HTTP pipeline (entire bandwidth), which rarely occurs, especially in cellular networks [26]. Although a client may probe the bandwidth share during OFF-interval and adapt properly in wired networks [8], this is not the case for cellular networks with significant channel dynamics. In fact, the client probes the bandwidth share every once a segment whereas the cellular link fluctuates at a much smaller scale due to its small-scale fading. The channel condition may even suddenly degrade when the client just probes a very high data rate.

Second, *TCP is not necessarily fair in cellular networks*. It has been widely assumed that TCP can provide fair performance for multiple flows. However, measurement studies over a 3G network [9] has shown that the fairness among multiple competing TCP flows fluctuates significantly when the last-hop cellular link is the bottleneck, i.e., TCP can achieve rather unfair throughput (more than 5 times difference) in cellular networks. The major cause behind this phenomenon is the poor interaction between TCP congestion control and proportional fair (PF) scheduling in cellular networks. Since there is a mismatch between the time scale of these two scheme that both seek to achieve certain fairness for multiple clients, it might be sometimes difficult to make their impacts synchronized. For example, TCP may plan to increase the sending rate of a target flow to approach fairness at a large time scale (multiple of RTTs), but the PF algorithm would fairly allocate channel resources to all the flows in a small time scale (every 2ms). This makes the target flow highly difficult to increase its throughput and reach fairness, especially when the number of competing flows are large. Therefore, existing HAS schemes [8], [11] depending on TCP fairness are insufficient to accomplish fair performance for multi-client cellular networks. Instead, it is imperative to design an adaptation framework on top of TCP in order to guarantee fairness in cellular networks.

Third, *client-driven adaptation is insufficient for QoE fair-*

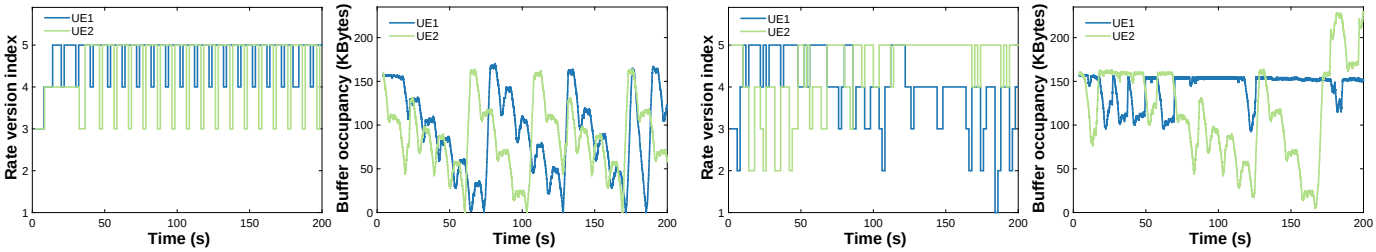


Fig. 2. a) Requested bitrate version and b) buffer occupancy versus time without CQI information; c) requested bitrate version and d) buffer occupancy versus time with CQI information.

ness. Current client-driven schemes address the fairness by handing down this task to the underlying TCP while the application layer aims to maximally approach the bandwidth share. However, as TCP fairness is not that reliable in cellular networks, the obtained bandwidth share may be highly diverse for different clients. In this case, a client with better QoE may even need to request a bitrate lower than the available bandwidth in order to save some channel resources. That way, the bandwidth share of other coexistent clients with worse QoE can be increased, which allows those clients to stream a higher bitrate video and accordingly enhances the QoE fairness. Nevertheless, an individual client will not be able to decide whether or not its bitrate should match the bandwidth and how conservative its bitrate should be in order for the fair performance. This is due to the fact that a client is unaware of other clients' bandwidth share or channel condition. Such a rate optimization on top of unfair bandwidth is only possible when the information of multiple clients are available and jointly considered [12], [13].

B. Motivational Study

Next, we explore the aforementioned issues and confirm the instability and unfairness in cellular networks via simulation studies.

We consider a HAS system over cellular networks as shown in Fig. 1. Each client periodically measures per-segment end-to-end throughput and requests the highest rate that can be supported by the measured throughput. For the simplicity of illustration, we run a toy example with two users streaming a video of 5 bitrate versions by ns-2 simulator. We implement a standard-compliant 3.5G High Speed Packet Access (HSPA) networks as the underlying cellular networks [27]. The results of requested bitrate index (greater index, higher bitrate) versus time is shown in Fig. 2a. It is clear that the bitrate request oscillates for both users. This is attributed to the mismatch between throughput and highly dynamic bandwidth in cellular networks. Since the client is oblivious to such a shared channel, it would overestimate the bitrate selection, which accordingly causes congestion when streaming the next segment. The congestion in turn brings a lower throughput and lower selected bitrate. Eventually, the rate oscillations shown in the figure would occur. Furthermore, as shown in Fig. 2b, such oscillated

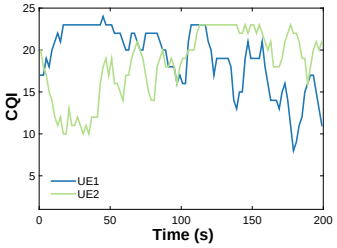


Fig. 3. Link condition versus time.

bitrate requests will definitely trigger frequent re-buffering in the player and degrade users' QoE. Based on this result, we confirm that conventional client-side HAS also suffers from playback instability in cellular networks.

We now study the impacts of available radio channel information on video playback. Channel quality indicator (CQI) is a periodically computed RAN metric. It can be used to estimate the maximum transmission rate of the downlink radio channel by using the look-up table in 3GPP standard [28]. In this simplified evaluation, we assume that the clients can access the CQI to estimate the maximum transmission rate as the upper bound of the rate selection. We re-run the HAS system and show the results of bitrate requests and buffer occupancy in Fig. 2c and 2d, respectively. We observe that the rate oscillations are somewhat mitigated and there is no re-buffering events for both users. Besides, the rate selection is generally consistent with the CQI trend shown in Fig. 3. For example, the channel condition of user 1 is relatively bad in the second half of the evaluation and its selected rate is accordingly low during this period. These results are reasonable because even though there may be overlapped ON-OFF pattern for the clients, CQI places a constraint on the bitrate adaptation. However, we also notice that the bitrate of two clients is not as fair. We believe this results from the distinct channel condition and the unfair bandwidth share of the clients in cellular networks. In fact, the clients essentially follow lower-layer channel variation of themselves. If the clients are aware of each other's situation, they may be able to strike a balance between maximum rate/CQI and fair bitrate.

Based on these simple yet insightful results, we recognize that RAN information such as CQI can play an important

role in rate adaptation. It can largely mitigate the problems caused by inaccurate bandwidth estimation. Furthermore, a joint adaptation that leverages application-layer and RAN information of all the clients is very much needed in order to achieve enhanced QoE for individual users and QoE fairness among users in cellular networks.

C. Architecture Design Considerations

We now proceed to discuss whether we can address the above problems in the current HAS architecture and whether we need a new architecture tailored to mobile cellular networks.

Client-side adaptation has been widely implemented in modern internet streaming systems since it can easily be deployed without modifying current web servers or in-network routers. It also achieves satisfactory scalability and can reflect the condition of the local device. Although we can expect client-side adaptation to become the dominant strategy in wired and WiFi networks, it might incur critical problems in cellular networks. With highly dynamic cellular links and unfair underlying TCP, it is unlikely for a single client to capture the fair bitrate under current channel conditions all by itself. Therefore, it is intuitive and necessary to collect the information of multiple clients and jointly adapt the bitrates.

One possible solution is to invite the video servers to assist the video adaptation [12], [13]. Nevertheless, this requires new central units in the servers and extra deployment of CDN to coordinate the strategy, which is beyond current commodity web servers designs. Alternatively, we can shift the adaptation intelligence to the base station and allow a central controller to jointly optimize the bitrate selection. This is in harmony with the centralized resource management framework in cellular networks without scalability and compatibility issues. Thanks to the computation-powerful edge cloud that can access the radio channel information, such a joint adaptation using low-layer information becomes certainly feasible. This strategy should work satisfactorily if the client device can physically support all the available bitrates. However, sometimes it is the client side that bounds the bitrate adaptation, e.g., due to limited display resolution and power support. Hence, it is still essential to consider the constraints of client-side characteristics in order to visualize the entire video streaming pipeline.

In this research, we propose Prius, a hybrid edge cloud assisted client-driven rate adaptation framework that jointly optimizes the bitrates of multiple clients at the edge cloud while taking into account the client-side capabilities. Note that Prius is not a design that shall completely replace the standard client-side HAS strategy. Instead, we provide it as a complimentary choice for HAS over cellular networks that is compatible with current cellular and adaptive streaming ecosystems.

IV. Prius: Architecture

Prius concentrates on the system architecture as shown in Fig. 1. The HAS server stores pre-encoded videos that all have M bitrate versions. Each version of video is characterized by video bitrate r and is split into multiple video segments with the same segment length. We focus on HAS over a single mobile cell, where a set of users $\mathcal{N}$ are subscribing to HAS services and each user is indexed by $i, i = 1, 2 \cdots N$. We assume the cellular entities, e.g., the cellular scheduler, operate in the same way as conventional mobile networks. The edge cloud can be deployed by the service provider and/or the mobile operator at the base station in order to enhance the quality of mobile services.

Prius consists of a Prius client and a Prius adaptation module for the hybrid adaptation. Since edge cloud is computationally powerful and can access RAN information available in the base station, e.g., CQI, Prius overlays a layer of adaptation intelligence at the edge cloud on top of individual clients' bandwidth-irrelevant bitrate requests. In particular, as it is unlikely to accurately capture the unfair bandwidth share in cellular links, Prius clients request a bitrate without concerning the bandwidth. However, all other device related constraints, such as display, computation ability, and battery power, are considered. This client request is in fact a upper bound of the bitrate request br_{bound}, which reflects the maximum bitrate supported in the Prius client. At the edge cloud, the Prius adaptation module can explore the channel knowledge of multiple streams for joint adaptation while meeting the local client-side requests.

The operation process of Prius is proceeded as follows. Initially, the HAS server sends out the MPD so that Prius adaptation module and clients will have the knowledge of available video representations. At each adaptation period that equals to the segment length, Prius clients request a video segment at a certain bitrate version based on its local bandwidth-unrelated factors. The request can be recognized as a naive suggestion for the edge cloud, which means no local hardware or operating system adjustment is needed. In other words, there would be no conflict if the client request is modified later. Unlike conventional client-side adaptation where the cellular networks simply forward the client requests to the video server, the edge cloud will intercept the requests. Prius adaptation module will then overwrite the adaptation decisions based on client-side request br_{bound} and cell-wide optimization of multiple clients, where both low-layer CQI and high-layer playback information are utilized. Client playback information for adaptation, such as buffer and QoE status, is embedded in the periodic feedback from clients. This is feasible as 3GPP has standardized the quality metrics reporting process for clients and uses HTTP POST as the reporting protocol [29].

The bitrate adaptation results are then delivered to the video server for streaming the next segment. In this way, the users are able to enjoy the video with optimized and fair QoE while no modification is needed in the video servers. Through this design, the proposed architecture can be also implemented friendly without modifying current infrastructure of cellular schedulers and the standard request-response framework of HAS.

V. PRIUS: DESIGN MODELS

In this section, we introduce the design models used in Prius. By taking full advantage of the new capacity of edge cloud, Prius presents two designs that facilitate the hybrid adaptation strategy within the framework, i.e., QoE continuum model and exponential smoothing based channel estimation.

A. QoE Continuum Model

In this section, we introduce a recently developed quality metric, QoE continuum [30], as the objective measure for optimizing the proposed hybrid rate adaptation.

Psychological research have discovered that the strength of human memory decays exponentially with respect to time [31]. For example, the visual perception of most recent video frames plays a more important role toward forming one's subjective QoE than that of less recent frames. Such effect has been suggested by ITU standard [32] to be applied in continuous quality evaluation. Hence, we exploit this effect to model the QoE continuum and measure QoE in a timely continuous fashion. We first assume only one frame can be played at one moment. In our previous study [30], we have carefully analyzed such effects and derived Q_k, the QoE continuum at moment k, as the weighted summation of instantaneous user experience over all previous moments until the measuring moment, i.e.,

$$Q_k = \gamma Q_{k-1} + (1-\gamma) q_k \tag{1}$$

where k ($k > 0$) is a index indicating a certain moment, q_k is the instantaneous user experience at moment k, Q_{k-1} is the QoE continuum at the previous moment $k-1$, and γ is the characterization constant of the memory strength. Thus, with a given initial QoE Q_0, we could iteratively fit in each instantaneous experience $(q_1, q_2, \cdots q_k)$ and finally output the QoE continuum at moment k. Note that Q_k, $|q_k|$ and γ all belong to (0,1]. With this formulation, we can capture the QoE from all previous moments until the current measuring moment.

We now proceed to introduce the computation of q_k. At a particular displaying moment of HAS systems, the video player can either normally playback a frame or freeze at a certain previous frame due to streaming irregularity. We disregard frame artifacts caused by transmission distortion since HAS is virtually loss-free due to the underlying TCP mechanisms. Therefore, the instantaneous user experience q_k at moment k can be expressed as two cases,

$$q_k = \begin{cases} q_{k,play} & \text{if playing normally at moment } k \\ q_{k,stall} & \text{if stalled at moment } k \end{cases} \tag{2}$$

For the moment with normal playback, the instantaneous user experience is dictated by the image quality of that frame. Thus we can predict image quality from the quantization parameter (QP) using a linear model and estimate the instantaneous user experience $q_{k,play}$ as

$$q_{k,play} = aQP_k + b \tag{3}$$

where a and b are video content-specific parameters.

When the bitrate of the streamed video exceeds the available bandwidth and the selected video has not been downshifted to a lower quality, playback interruption may occur due to the client re-buffering. In this case, the video player's screen will stall at the most recently displayed image and consequently the user will undergo certain loss of expected visual information. Thereby, we model the instantaneous user experience for a stall moment $q_{k,stall}$ as the visual information loss L_k at moment k scaled by the current user expectation E_k [30], i.e.,

$$q_{k,stall} = -L_k E_k. \tag{4}$$

We signify E_k by the playback quality of the most recently displayed frame. This is because the higher quality a user is previously enjoying, the higher current expectation he/she should have. We assume a frame is most recently playbacked at moment j ($j < k$). Thus we have $E_k = q_{j,play}$. Note that j is not necessarily equal to $k-1$ because one can suffer the playback stall for a long time before moment k. Besides, the visual information loss L_k should measure the mean squared error (MSE) distortion between the currently expected frame and the most recently displayed frame. We denote the original non-compressed frames of the video in pixel domain by a vector $\mathbf{f}$ and the decoded frames by $\mathbf{f}'$. We assume the expected frame to play at moment k is the gth frame of the video and thus the most recently displayed frame at moment j is the $(g-1)$th frame. Therefore, we have

$$L_k = \mathbb{E}\|\mathbf{f}(g) - \mathbf{f}'(g-1)\|^2 = \mathbb{E}\|\mathbf{R}_g\|^2 \tag{5}$$

where $\mathbf{R}_g$ is the residual signal between the previously decoded frame and the currently expected frame. Note that residual signal is usually utilized in video coding process for inter-frame prediction. We have found in extensive experiments [30] that $L_k = \mathbb{E}\|\mathbf{R}_g\|^2$ can be linearly approximated by $\frac{bc_{j,play}^2}{\eta}$, where $bc_{j,play}$ is the bit count of the last displayed frame at moment j and η is the compression ratio. In rate distortion theory, distortion is commonly modeled as a logarithmic function in terms of bitrate. Hence, we apply a logarithmic operator to the bit count and then bound L_k in (0,1] as follows,

$$L_k = \frac{\log(\min(bc_{j,play}, bc_{\text{QP}})) + d}{\log(bc_{\text{QP}}) + d} \tag{6}$$

where bc_{QP} is the upper bound of frame size by a specific QP, and $d = -\log(\eta)/2$ is a model constant. The value of bc_{QP} can be calculated online based on previous streaming information of decoded bits. Details on deriving (1)-(6) and subjective tests to validate the model can be found in [30].

We now explain how the QoE continuum model can also effectively characterize playback smoothness, which is another key factor that impacts the user experience. Suppose one has been enjoying the video with decent quality for a while (e.g., $Q_{k-1} = 0.9$), but the bitrate of the upcoming segment is abruptly downgraded due to bandwidth decrease of the mobile networks. This will definitely lead to annoying user experience. By using (1), we can infer the same effect. When viewing the low-quality frame k (e.g., $q_k = q_{k,play} = 0.6$), the value of Q_k would decrease. Such decrease of Q would be accumulated until the end of the entire segment. The larger the difference

between Q_{k-1} and q_k, the smaller the value of Q and the worse QoE continuum would be. In other words, the computation of Q inherently takes into account playback smoothness by placing a penalty on abrupt quality changes.

B. Channel Estimation

To perform channel-aware rate adaptation, it is important for Prius to accurately estimate the maximum transmission rate in the upcoming adaptation period.

1) Extracting a Channel Metric: Although periodically reported CQIs are available at the edge cloud for estimation, the number of CQIs are too numerous to conduct effective exploration. For example, CQI is reported every 2 ms in HSPA networks, which results in 1000 values for a 2-second HAS segment. Instead of including all available values and CQI history into the estimation, we believe it is better to use properly pooled CQI statistics to reduce the computational complexity. Hence, we propose to adopt the average CQI within an adaptation period to characterize the maximum transmission rate R. In other words, we use the rate indicated by the average CQI to approximate the real rate computed from all individual values of CQI, i.e.,

$$R_t = \frac{f(CQI_{avg,t})}{\tau} \approx \frac{\sum_\tau f(CQI_\tau)}{\tau N_\tau} \tag{7}$$

where $f()$ is the 3GPP mapping table that determines the maximum transmission bits size during a CQI report moment τ, $CQI_{avg,t}$ is the average CQI of the period t, and N_τ is the total number of such moments within one adaptation period. The reason of this choice is that the mapping table $f()$ is a near-linear increasing function of CQI, i.e., higher CQI indicates higher transmission rate. Thus the mapped rate of the average CQI is expected to be very close to the average of all the rates mapped from each CQI.

We plot the real maximum transmission rate over 100000 CQIs and the approximated rate using average CQI per segment in Fig. 4. It can be seen that the approximated rate closely matches the real rate under both slow moving (3 km/h) and fast moving (120 km/h) environment. Therefore, we conclude that using average CQI to extract the channel condition of an adaptation period is indeed sufficient.

2) Estimating the Channel Condition: We now introduce how to exploit the average CQI of current period in order to estimate the CQI of the upcoming period. Simply using the current CQI as the estimated upcoming CQI could be highly inaccurate because the channel conditions of two consecutive adaptation periods can be significantly different, especially under fast-changing channels. Instead, it is more reasonable to leverage both the CQI history and current CQI in order to interpret the trend of channel variation. We expect that such a pooled estimate is able to capture the channel characteristics more accurately.

We propose a single exponential smoothing based scheme to estimate the CQI of the upcoming adaptation period. Single exponential smoothing has been commonly used in the forecasting of time series data without systematic trend [33], which is especially true for the highly dynamic radio channel

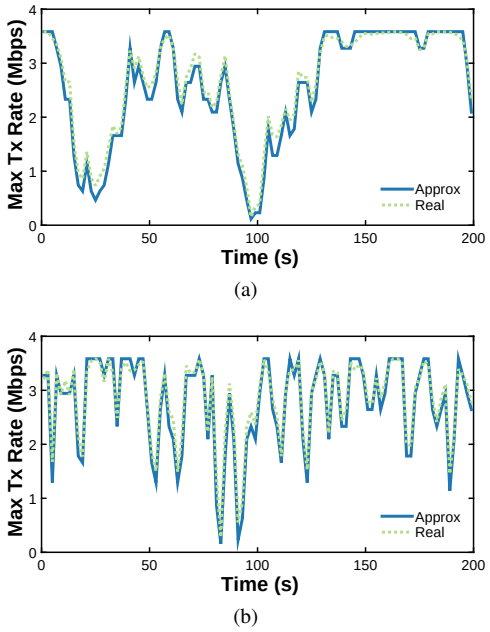


Fig. 4. The approximated transmission rate from average CQI and real transmission rate under a) slow moving status; b) fast moving status.

dictated by user movement, signal fading, shadowing, etc. The estimated CQI of the upcoming period t', $CQI_{est,t'}$, can be expressed as,

$$CQI_{est,t'} = \alpha CQI_{avg,t} + (1-\alpha) CQI_{est,t} \tag{8}$$

where α ($0 < \alpha \leq 1$) is the smoothing factor. Since the optimal α shall depend on channel variation pattern and such patterns could be largely distinct under different moving status, it is critical to experimentally study the impacts of α. Details of the choice for α will be discussed in Section VIII.

It is important to note that more complicated estimation techniques may be able to achieve higher accuracy. However, they usually require more parameters, which significantly complicates the process of parameter tuning and determination. We choose single exponential smoothing in order to strike a trade-off between estimation accuracy and implementation complexity.

Based on the estimated CQI, the edge cloud can finally obtain the estimated maximum transmission rate of the upcoming period via the 3GPP mapping table, i.e.,

$$R_{t'} = \frac{f(CQI_{est,t'})}{\tau} \tag{9}$$

VI. Prius: Hybrid Rate Adaptation Problem

We have introduced the architecture of Prius, as well as its major components for measuring QoE and estimating channel status. With these component designs, the Prius adaptation module of the framework can now be introduced in this section.

A. Problem Formulation

The adaptation module in the edge cloud is designed to solve the QoE CONTINUUM ADAPTATION PROBLEM (QCAP), which specifically addresses the QoE continuum issue and fairness issue for multiple clients in resource-limited cellular networks. We formulate QCAP as follows.

*Definition 1 (*QCAP*):* Suppose the QoE continuum model in (1) and the channel estimation in (7)-(9) are adopted. Given a video, segmented into T-second chunks, with a set of different bitrate versions $\mathcal{V} = \{br_1, br_2, \ldots, br_{\text{M}}\}$ and N clients with client-bounded bitrate $br_{i,bound}$, each with current QoE continuum Q_i, current buffer status B_i, and estimated maximum transmission rate R_i, the problem is to determine the bitrate version r_i ($r_i \in \mathcal{V}$) for all clients $i \in \mathcal{N}$ such that the average QoE continuum for all users at the end of next adaptation period is maximized without exceeding shared resource constraint and client-side bound.

Mathematically, QCAP can be written as,

$$\begin{array}{ll} \max_{(\mathbf{r})} & \frac{1}{N}\sum_{i=1}^{N} Q_{i,t+T} \\ \text{s. t.} & \sum_{i=1}^{N} \varphi_i \leq 1 \\ & r_i \leq br_{i,bound} \\ & Equation\ (1)-(9) \end{array} \tag{10}$$

where $\varphi_i = \frac{r_i}{R_i}$ is the radio resource share of client i and equation (1)-(9) define the computation of QoE continuum and estimated maximum transmission rate.

The adaptation wisdom behind (10) is that higher bitrate version is generally assigned to those users who currently possess a lower QoE continuum value and a better channel condition, while significant bitrate variation shall also be avoided. For example, when two users are currently enjoying the same channel condition and their QoE continuum is 0.8 and 0.5, respectively, higher bitrate video (e.g., $q_k = 0.9$) is assigned to the user with a lower current QoE continuum because such adaptation will result in maximum increase in the average QoE continuum, i.e., $[0.5\gamma + 0.9(1-\gamma)] - 0.5 > [0.8\gamma + 0.9(1-\gamma)] - 0.8$. In other words, when a user enjoys good experience for a long time, his/her satisfaction will rise less than the one with bad previous experience if the video quality can be raised. Consequently, we can enhance not only the QoE continuum but also the fairness of users. Additionally, the penalty on playback variations can avoid the sudden big change and keep the playback smooth.

B. Complexity Analysis

It is necessary to carry out complexity analysis for a real-time optimization problem such as QCAP before studying the solution algorithms. We propose to analyze the complexity of QCAP by first considering the following simplified version of QCAP (QCAP-SV).

*Definition 2 (*QCAP-SV*):* Suppose that the same QoE continuum model, channel estimation scheme, and video input are adopted as QCAP. However, we are given that $br_{i,bound} = br_{\text{M}}$ and that the QoE continuum $Q_{i,t+T}$ at next switching point and the resource share $\varphi_i = \frac{r_i}{R_i}$ when streaming all M versions of video for all N clients have been pre-computed. The objective and constraint shall remain the same.

Theorem 1: QCAP-SV is NP-hard.

Proof: We will demonstrate that PARTITION PROBLEM, which is a well known NP-complete problem, can be polynomially reduced to QCAP-SV.

*Definition 3 (*PARTITION PROBLEM*):* Given a set of n integers $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ such that $\sum_{i=1}^{n} x_i = 2D$, determine a subset $\mathcal{X}' \subseteq \mathcal{X}$ such that $\sum_{x_i \in \mathcal{X}'} x_i = \sum_{x_i \in \mathcal{X} - \mathcal{X}'} x_i = D$.

For an arbitrary instance of PARTITION PROBLEM, we construct an instance of QCAP-SV, where we have $M = 2$ bitrate versions for a video and $N = n$ clients. One integer x_i corresponds to one client i. We set $\varphi_i = Q_{i,t+T} = \frac{x_i}{D}$ for a client watching the video with br_1 while for a client enjoying the video with br_2 we set $\varphi_i = Q_{i,t+T} = 0$. Clearly, the construction algorithm can be completed in polynomial time.

If the instance of QCAP-SV is an optimal solution, we shall have a subset of the clients $\mathcal{C}'$ selecting br_1 while the others selecting br_2 such that $\sum_{i \in \mathcal{C}'} \varphi_i = \sum_{i \in \mathcal{C}'} \frac{x_i}{D} = 1$ and the maximum value of the objective is also 1. Thus we can find a solution of PARTITION PROBLEM by grouping those x_i corresponding to the client i selecting br_1.

Conversely, if there is a solution for PARTITION PROBLEM such that $\sum_{x_i \in \mathcal{X}'} x_i = \sum_{x_i \in \mathcal{X} - \mathcal{X}'} x_i = D$, we will have $\sum_{x_i \in \mathcal{X}'} \frac{x_i}{D} = 1$. By selecting br_1 for those clients i corresponding to integers in $\mathcal{X}'$ and br_2 for the others, we can achieve the maximum average QoE continuum while satisfying the resource constraint.

Therefore, we can prove that PARTITION PROBLEM is polynomially reducible to QCAP-SV and therefore QCAP-SV is NP-hard. ■

Since QCAP-SV is a simplified version of QCAP by pre-computing all possible $\frac{r_i}{R_i}$ and $Q_{i,t+T}$ and by fixing M video versions for N clients, this follows that QCAP is also NP-hard.

VII. PRIUS: SOLUTIONS TO HYBRID ADAPTATION

A. Computing QoE Continuum

To solve QCAP, it is important to clarify how Prius adaptation module at edge cloud would compute the QoE continuum at the next switching point $Q_{i,t+T}$ by (1).

When a video rate version r_i is considered, the adaptation module in the edge cloud is able to estimate the QoE continuum at the next switching point $Q_{i,t+T}$ by using estimated maximum transmission rate R_i, current QoE continuum Q_i and buffer status B_i. $Q_{i,t+T}$ can be recursively calculated according to (1) by simulating whether the player is playing or stalling at each display moment during the T-second period.

If $B_i > 0$, client i is not re-buffering and will normally playback the buffered data and newly requested data. Then $Q_{i,t+T}$ will be iteratively computed from Q_i by using $q_{k,play}$. For $k \in [t, t+B_i]$, the buffered frames are played while the newly requested frames (bitrate r_i) are played when $k \in [t+B_i+1, T]$.

On the other hand, if $B_i = 0$, the player needs to re-buffer the newly requested segment until B_i reaches the playback threshold. After that, it can normally playback the requested segment. With selected bitrate r_i, $Q_{i,t+T}$ will be iteratively computed by first using $q_{k,stall}$ for $k \in [t, t']$ and then $q_{k,play}$

for $k \in (t', T]$. Note that t' can be computed because the transmission rate, requested bitrate and playback threshold are all known. This way, the edge cloud can accurately estimate the QoE continuum at the next switching point and solve for QCAP accordingly to obtain the most satisfactory and fair adaptation decision.

B. Optimal Dynamic Programming Algorithm

We now propose a dynamic programming algorithm to optimally solve for QCAP. The optimal results shall be served as the upper bound for the QoE continuum as well as a guideline for developing a practically efficient heuristic algorithm.

Initially, we obtain $\mathcal{V}_i = \{br_1, br_2, \ldots, br_{i,bound}\}$ and compute $\varphi_i(r_i)$ and $Q_{i,t+T}(r_i)$ for all clients $i \in \mathcal{N}$ when they select all possible bitrates $r_i \in \mathcal{V}_i$. We then derive the remaining algorithmic steps by first considering a sub-instance with N' $(1 \leq N' \leq N)$ clients and the total resource c $(0 \leq c \leq 1)$. Let $F(N', c)$ denote the optimal value of its objective. We assume $F(N', c) = -\infty$, if there is no feasible solution for the sub-instance. We denote the minimum possible resource share of client i as $\varphi_i^* = \min\{\varphi_i(r_i) | r_i \in \mathcal{V}_i\}$. If there is only one client, we clearly have

$$F(1,c) = \begin{cases} -\infty & c = 0, \frac{1}{\Delta}, \ldots, \varphi_1^* - \frac{1}{\Delta} \\ \max\{Q_{1,t+T}(r_1) | & \\ r_1 \in \mathcal{V}_1, \varphi_1(r_1) \leq c\} & c = \varphi_1^*, \varphi_1^* + \frac{1}{\Delta}, \ldots, 1 \end{cases} \tag{11}$$

where $\frac{1}{\Delta}$ is the smallest unit for resource share. Similarly, when $N' = 2, 3, \ldots, N$, there are also two cases. If the total resource c is less than the minimum resource share for N' clients, the total QoE continuum would be $-\infty$. Otherwise, the optimal value would be the sum of the maximum QoE continuum for $N' - 1$ clients with $c - \varphi_{\mathrm{N}'}$ resource and the QoE continuum for the N'th client, i.e.,

$$F(N',c) = \begin{cases} -\infty & c = \mathcal{S} \\ \max\{F(N' - 1, c - \varphi_{\mathrm{N}'}(r_{\mathrm{N}'})) & \\ +Q_{N',t+T}(r_{\mathrm{N}'}) | r_{\mathrm{N}'} \in \mathcal{V}_{\mathrm{N}'}, \varphi_{\mathrm{N}'}(r_{\mathrm{N}'}) \leq c\} & c = \mathcal{S}' \end{cases} \tag{12}$$

where $\mathcal{S} = \{0, \frac{1}{\Delta}, \ldots, \sum_{i=1}^{N'} \varphi_i^* - \frac{1}{\Delta}\}$ and $\mathcal{S}' = \{\sum_{i=1}^{N'} \varphi_i^*, \sum_{i=1}^{N'} \varphi_i^* + \frac{1}{\Delta}, \ldots, 1\}$. By substituting $N' = N$ and $c = 1$, we can easily obtain the optimal solution for QCAP.

To derive the time complexity of the optimal dynamic programming algorithm, we need to consider both initialization and the subsequent steps. In this case, the initialization takes $\mathcal{O}(N|\mathcal{V}_i|) = \mathcal{O}(NM)$ time. Besides, since $F(N, 1)$ depends on $F(N-1, 1)$, we need to compute all N sub-instances, each of which would take $\mathcal{O}(M\Delta)$ time. Therefore, the total time complexity for the dynamic programming algorithm is $\mathcal{O}(NM\Delta)$. Since Δ can be a huge number, this algorithm could be in exponential time in terms of the size of the input.

C. Efficient Heuristic Algorithm

In this section, we propose a tree-pruning like heuristic algorithm to efficiently solve for QCAP and approximate the optimal solution. The basic idea is to prune the channel resources of each user in such a way that the overall QoE continuum can be maximized while the resource sharing constraint is satisfied. In other words, we aim at achieving the maximum possible QoE continuum by appropriately deallocating the resource from clients until the resource budget is met.

At the initialization stage, the resource share $\varphi_i(r_i)$ and QoE continuum $Q_{i,t+T}(r_i)$ for all N clients with all bitrate $r_i \in \mathcal{V}_i$ are calculated. We also assume the initial video version for all N clients are the highest possible bitrate version. We define *deallocating slope*, denoted by δ_i, as the variation of QoE continuum per unit resource when client i adapts to a lower bitrate video, i.e.,

$$\delta_i(r_i, r_i') = \frac{Q_{i,t+T}(r_i) - Q_{i,t+T}(r_i')}{\varphi_i(r_i) - \varphi_i(r_i')}, \quad \{r_i' | r_i' \in \mathcal{V}_i, r_i' < r_i\} \tag{13}$$

Note that a larger δ indicates more QoE loss if degrading the video bitrate and vice versa for a smaller δ. For the special case of $\delta < 0$, it indicates that the degradation of video bitrate actually increases, rather than decreases, the QoE continuum. This is possible since streaming a high bitrate video that exceeds the available bandwidth may cause frequent playback interruption, which would negatively impact the QoE. The proposed algorithm is summarized in Algorithm 1.

Algorithm 1 Tree-pruning Heuristic Algorithm

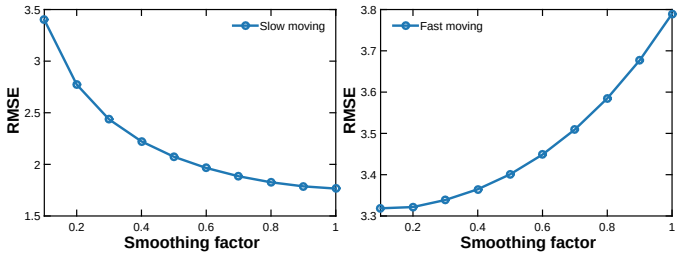
```
procedure RATE ADAPTATION
    Compute φ_i(r_i) and Q_{i,t+T}(r_i), ∀i ∈ 𝒩, ∀r_i ∈ 𝒱_i
    r_i ← br_{i,bound}, ∀i ∈ 𝒩
    𝒩' ← 𝒩
    while Σ_i φ_i > 1 & ∃i, r_i ≠ br_min do
        for i ∈ 𝒩' do
            δ_{min,i} ← min{δ_i(r_i, r_i') | r_i' ∈ 𝒱_i, r_i' < r_i}
        end for
        (i*, r*) ← arg min{δ_{min,i} | i ∈ 𝒩}
        𝒩' ← {i*}, r_{i*} ← r*
    end while
    return r
end procedure
```

At the first step, if the resource constraint has already been met, we output the highest bitrate $br_{i,bound}$ for all clients. Otherwise, at each subsequent step, we compute the deallocating slopes for all possible video degradation scenarios of all clients. We then find the client i^* with a reduced bitrate r_i' that accomplishes the smallest slope $\delta_i(r_i, r_i')$ and downshift client i^* to that optimal bitrate r^*. The resource budget is checked again to see if we need to continue such a pruning. The pruning and the checking process will repeat until the total resource is less than 1 or all clients' video are degraded to the lowest bitrate $br_{\min}$.

The time complexity can be measured as follows. Algorithm 1 takes $\mathcal{O}(N|\mathcal{V}_i|) = \mathcal{O}(NM)$ time to pre-compute $\varphi_i(r_i)$ and $Q_{i,t+T}(r_i)$ as in the optimal dynamic programming algorithm. In the worst case, the video bitrate always degrades just one level at each step for every client and the algorithm stops when the videos for all the clients degrade to the lowest bitrate. Specifically, a client needs to go through $M - 1$ slopes when

TABLE I. CELLULAR NETWORKS CONFIGURATION

Parameter	Value
Cell layout	Single hexagonal cell
UE distribution	Uniform
UE distance	250 $\sim$ 500 m
Path loss model	ITU Pedestrian and Vehicular [34]
UE speed	3 km/h and 120 km/h
HAS UEs	4
Transmission time interval	2 ms
UL/DL duplexing	TDD
BS transmission power	38 dBm
BS antenna gain	17 dBi



(a) (b)

Fig. 5. Root mean squared error for the estimated CQI and real CQI under a) slow moving status; b) fast moving status.

downshifting the rate one level from the initial maximum rate. Then $M-2$ slopes need to be checked for another level of degradation. For downgrading all N clients to the minimum rate, it would involve

$$N(M-1+M-2+\cdots+1)=N\frac{M(M-1)}{2} \qquad (14)$$

times of slop calculations and comparisons. Since each pari of slop calculation and comparison takes a constant time, we need $\mathcal{O}(NM^2)$ time to complete the entire algorithm in the worst case. Hence, the overall time complexity for the heuristic algorithm shall be in polynomial time, i.e., $\mathcal{O}(NM^2)$.

VIII. PERFORMANCE EVALUATIONS

To verify the performance of Prius, we have built a ns-2 based environment that includes video servers, core networks, cellular networks, the edge cloud with Prius adaptation module and Prius clients, based on the architecture in Fig. 1. The HAS server stores five test sequences "Big Buck Bunny", "Stefan", "Tears of Steel", "Crowdrun" and "Akiyo". We encode the source sequence into H.264/AVC video and the video bitrate of 5 versions ($br_1, br_2, \cdots$) ranges from 70 Kbps to 700 Kbps. The segment length T is 2 seconds and the frame rate is 30 fps. Clients will randomly request another sequence when one sequence ends.

We assume all the client-side bitrate bounds are highest bitrate. We also implement a 3.5G HSPA networks as the underlying cellular networks by using EURANE [27], which is a open-source implementation mainly developed by Ericsson and can generate industry-compliant RAN systems. The moving speed of users is 3 km/h under slow moving status (ITU pedestrian model) and 120 km/h under fast moving status (ITU vehicular model). Detailed configuration of the mobile cellular networks is shown in Table I. Other wired network parameters are set to the default values as in EURANE.

Regarding the parameters of QoE continuum model, we inherit them from [30], wherein the accuracy of the model is validated by both objective and subjective tests. The memory strength γ is set to be 0.71. The instantaneous user experience q_{play} of different versions is computed based on the bitrate of a video and its corresponding QP. For example, q_{play} belongs to $\{0.85, 0.88, 0.92, 0.94, 0.95\}$ for "Stefan" video. Since the initial buffering can be regarded as a special case of playback stalling, the instantaneous user experience during initial buffering is calculated using (4) with constant $L_{ini}=-0.5$. The initial version is the medium bitrate (br_3) for all users. The simulation runs 200 seconds.

We compare the performance of Prius with several representative reference systems. To highlight the advantages of edge cloud assisted hybrid adaptation, we first implement a conventional *client-side* rate adaptation algorithm, which requests the maximum bitrate that can be supported by current per-segment throughput. It essentially means that the edge cloud would simply forward the rate selection from clients without any overwriting. We also implement a typical centralized adaptation algorithm (referred as *Instant*) that captures the logic behind many existing works, where the joint adaptation maximizes the utility function dictated by selected instant bitrate, subject to the channel constraint. One example of Instant-like algorithms is reported in [35], in which the utility is the MOS mapped from bitrate. Such adaptation focuses on enhancing streaming rate rather than QoE and does not consider fairness. Finally, we implement the optimal dynamic programming algorithm in Section VII-B to study the performance gap between the proposed heuristic algorithm and the theoretical upper bound.

A. Channel Estimation Results

In this section, we report the performance evaluation of the channel estimation in Prius. Specifically, we investigate the impacts of the smoothing factor α in the single exponential smoothing based channel estimation, in order to improve estimation accuracy under different moving status. We collect CQI trace of 4000 seconds using the simulated HAS system. Half of them is under slow moving environment while the other half is under fast moving environment. We estimate the average CQI for the upcoming period based on that of current period by ranging α from 0.1 to 1 via (8). A greater α indicates more impacts from current CQI and less impacts from CQI history on the estimated CQI. Zero value of α is not allowed as it essentially ignores the new measurement and all the estimated CQI will always be the initial CQI.

We show in Fig. 5 the root mean squared error (RMSE) of the estimation against the real trace. We observe an interesting trend that a greater α makes the estimation more accurate under slow moving environment whereas less accurate under fast moving environment. This phenomenon can be explained

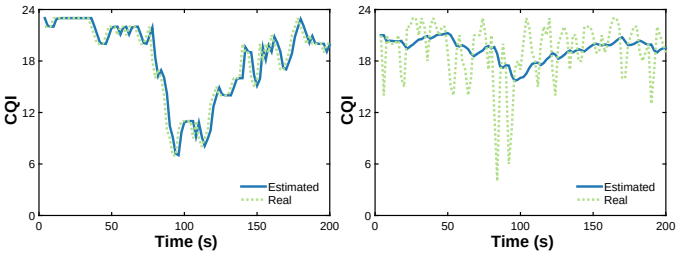


Fig. 6. The estimated CQI and real CQI of each period under a) slow moving status; b) fast moving status.

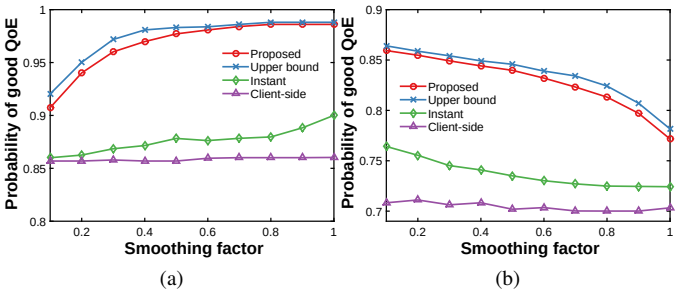


Fig. 7. Probability of good QoE versus α under a) slow moving status; b) fast moving status.

as follows. The channel variation of slow-moving users is relatively smooth and thereby the CQIs of two consecutive periods would be similar. By adopting a greater α, the estimation will be closer to the current measurement, which is able to quickly capture the variation while not smoothing out the trend. An example of real per-period CQI and estimated CQI by setting $\alpha = 1$ for a pedestrian user is shown in Fig. 6a. It demonstrates a close match between the real trace and estimated CQI.

On the other hand, the channel is usually varying abruptly and frequently under fast-moving environment. That is, the CQIs of two consecutive periods could be substantially different. If we repeat the current CQI as the estimated value ($\alpha = 1$), we may be moving away from the real value at every period because the variation is too fast that the upcoming CQI can take virtually any value. By setting a smaller α, the estimation can smooth out the unpredictable variation and grasp the general channel condition in a long-term scale. This will surprisingly improve the accuracy significantly. A example of CQI trace and estimation by setting $\alpha = 0.1$ for a vehicular user is shown in Fig. 6b. We can see that many random fluctuation are irrelevant to the previous CQI. In this case, a smoothed estimation could be more effective since even though the bandwidth is underestimated in one period it may be overestimated in the next period, making the estimated long-term bandwidth relatively accurate.

Based on the above analysis, we conclude that a greater α is recommended for slow-moving users whereas a smaller α should be adopted for fast-moving users.

B. QoE Continuum Results

We now report the procedures to evaluate the QoE continuum of the different HAS systems. We collect the actual QoE continuum (not the estimated $Q_{i,t+T}$) at all displaying frames from the video players. We consider the QoE continuum larger than 0.8 (corresponding to MOS 4.0) as "good" experience and show the probability of good QoE in Fig. 7. Under both slow and fast moving cases, Prius outperforms the reference systems. This is because Prius exploits the edge cloud to address the issue of shared bandwidth, which causes playback instability for case of the client-side adaptation that only uses end-to-end throughput based criterion. Furthermore, we formulate QCAP to jointly consider the QoE continuum and channel resources whereas *Instant* algorithm only aggressively maximizes the instant bitrate, which is not necessarily able to ensure QoE continuum. From these evaluations, we also observe that the proposed heuristic algorithm achieves a near-optimal performance, which validates the feasibility of the polynomial-time algorithm. Finally, the trend of QoE continuum versus α is complaint to the trend of channel estimation accuracy, i.e., more accurate channel estimation leads to more appropriate rate adaptation and thus better QoE continuum.

To investigate the contribution of different playback-related factors to the QoE continuum, we measure the following performance metrics [36] and report the results in Table II.

- Rate of Buffering (RoB): It represents the user average ratio of the number of re-buffering events over the length of the streaming session. RoB captures the frequency of playback interruption.
- Buffering Ratio (BR): It is computed by the user average duration of re-buffering divided by the total length of the streaming session, which indicates the general strength of playback interruption.
- Rate of Version Change (RoC): It signifies the user average number of bitrate version change per unit time, i.e., the frequency of rate variation.
- Level of Version Change (LoC): It is computed as the average level of bitrate change over all the bitrate changes of all users, which implies the average extent of each bitrate change.

It can be seen from Table II that Prius achieves better performance under almost all metrics. This is because the two important playback factors, re-buffering and smoothness, has been included in the proposed joint adaptation and QoE continuum framework. In contrast, the reference systems focus on maximally increasing the video bitrate. Furthermore, we observe that RoC has less impacts on QoE continuum than the other metrics. For example, although client-side adaptation changes bitrate least frequently, its QoE continuum is the worst. This is reasonable since rate variation does not have to be abrupt and smooth bitrate change is usually not annoying. However, frequent/long-term re-buffering and abrupt rate variation would easily interrupt users' immersion in viewing and degrade the QoE continuum.

We also demonstrate an example of playback bitrate for different systems in Fig. 8, where index zero represents the re-buffering. We can see that UE3 in all the schemes experiences

TABLE II. PLAYBACK PERFORMANCE ($\alpha = 0.5$)

Metrics	Slow moving				Fast moving			
	Proposed	Upper bound	Instant	Client-side	Proposed	Upper bound	Instant	Client-side
RoB	0.0026	0.0026	0.0065	0.0078	0.0052	0.0052	0.0129	0.0129
BR	3.5841	3.0627	17.6192	21.7239	10.8149	9.3409	23.8194	25.0299
RoC	0.0631	0.0618	0.0747	0.0168	0.0735	0.0722	0.0773	0.0168
LoC	-0.2312	-0.2301	-0.2569	-1.4583	-0.4350	-0.4220	-0.3721	-1.2500

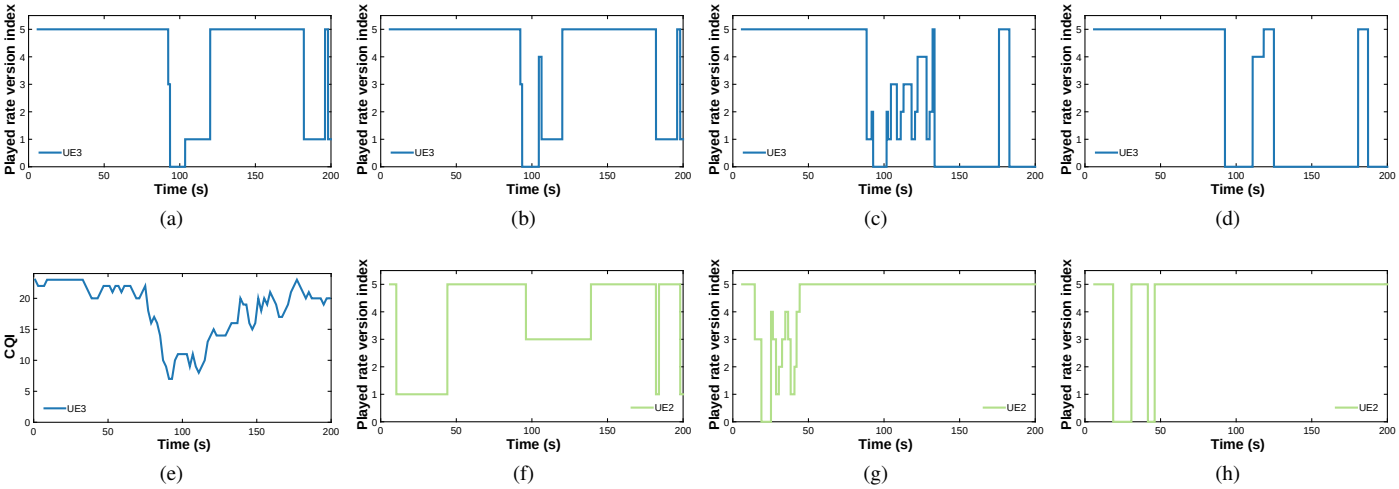


Fig. 8. An example playback a) bitrate of UE3 (upper bound); b) bitrate of UE3 (proposed); c) bitrate of UE3 (Instant); d) bitrate of UE3 (Client-side); e) CQI of UE3; f) bitrate of UE2 (proposed); g) bitrate of UE2 (Instant); h) bitrate of UE2 (Client-side).

a re-buffering and lower bitrate playback at around 100th second (Fig. 8a-8d) due to the sudden degradation of link CQI (Fig. 8e). Then the link CQI starts improving from the 120th second. In the proposed schemes (both upper bound and proposed algorithm), UE3 can successfully play bitrate #5 to quickly recover from the previous bad QoE. This is because the fair adaptation in the proposed algorithms decreases the bitrate of the other user, UE2, who has been enjoying a higher bitrate, in order to save some bandwidth share for UE3 (Fig. 8f). Note that since the results of upper bound algorithm is very similar to that of proposed algorithm, we only show the proposed algorithm for UE2. On the other hand, UE3 in Instant and Client-side algorithms also request bitrate #5 after the 120th second because their bitrate selection essentially follows the CQI increase. However, the bandwidth share of UE3 is not high enough and re-buffering of requested bitrate #5 was observed for both algorithms. This is because the two compared algorithms fail to consider fairness and UE2 in both schemes still selfishly streams the highest bitrate given their high CQI (Fig. 8g-8h). This leads to the congestion under the limited system capacity. Note that we do not include the results of UE1 and UE4 since both of them have a high CQI to support the playback of the highest bitrate in all the algorithms and thus have no impacts on the performance UE3 and UE2.

C. QoE Fairness Results

We compute the variance for probability of good QoE among all users in order to evaluate the QoE fairness. The results versus α under slow and fast moving environment is shown in Fig. 9. We observe that Prius shows near-zero QoE difference among the users and thus demonstrates the desired QoE fairness. In fact, the difference in the probability of good QoE among users is usually less than 0.05. This is because the proposed adaptation always assigns higher priority to users with previously bad experience. When the same high bitrate video is played, a user with bad previous experience would have a greater perception improvement than a user already having good experience. Nevertheless, the aggressiveness in reference algorithms to maximally capture throughput or channel variation may result in re-buffering due to the estimation error in throughput or channel condition. Therefore, the playback performance and QoE continuum among users can be significantly different.

IX. DISCUSSION

Server or Network Side HAS. It is true that HAS was originally developed such that the video client has the ultimate adaptation intelligence. This also simplifies the system-level implementation under current Internet infrastructure since only the client needs to be modified. However, in order to enhance QoE continuum, especially in multi-client wired/cellular networks, server or network side HAS may be more desirable in many cases. Such promotion of centralized HAS is also compatible with the current industry trend. For example, MPEG have included such ideas into its standardization process [37],

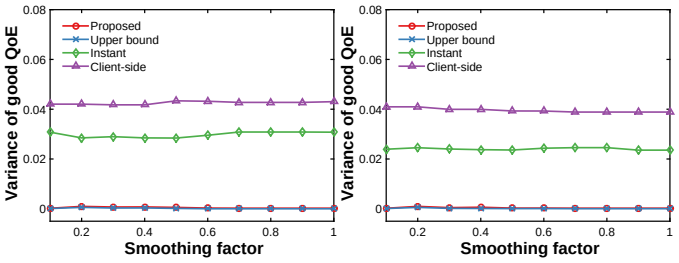


Fig. 9. Variance for probability of good QoE versus α under a) slow moving status; b) fast moving status.

which indicates that more auxiliary techniques will be defined to support such an architecture. Besides, content provider Netflix has recently also entered a peering agreement with network operator Comcast [38] to push its smooth streaming.

Other Bottlenecks than Cellular/Client. In this research, we implicitly assume that cellular links are the bandwidth bottleneck. By incorporating the device limitation from the client-side request, Prius can guarantee the end-to-end performance. In practice, backend CDNs can also be the bandwidth bottleneck. To address this issue, we can invite the client to send a standard bandwidth-based request based on per-segment throughput. If the client throughput inferred from the request is significantly lower than the previous bitrate selected by Prius which considers cellular bandwidth, there must be some congestion in the backend and the next request should be adjusted by the edge cloud accordingly. This calls for a future full-scale study on the algorithms and evaluation of backend detection. However, Prius architecture is compatible with this possible solutions and it is feasible to extend the current version of Prius to cater to the case of CDN congestion.

Channel Estimation. We provide the guideline of selecting the estimation parameter α through the evaluations under two extreme moving scenarios. Although we do not evaluate all moving scenarios in between, this can be achieved easily by adopting a similar experimental method to obtain the optimal α under different moving speed. A online algorithm that maps moving speed to optimal α can be developed in the edge cloud to carry out the online channel estimation with dynamic estimation parameter.

It is important to reiterate that this example channel estimation scheme is only designed to study the impacts of moving pattern on estimation accuracy. We have found that faster moving users need smoother channel estimation. Based on this key observation, more sophisticated estimation schemes, e.g., double/triple exponential smoothing, can be developed on top of the proposed scheme to achieve better estimation results.

X. Conclusion

In this paper, we have presented a new investigation of HAS over mobile cellular networks under the new context of edge cloud. We first demonstrate the issues of playback instability and unfairness of multiple competing users when only the client-side adaptation is implemented. Based on such observations, we design Prius, a new hybrid edge cloud and client adaptation framework for mobile cellular networks taking the full advantage of the new capacities with the emerging edge cloud. In order to explore the RAN and application-layer information inherent to the edge cloud, we developed a channel estimation scheme and a joint adaptation strategy with QoE continuum as the optimization criterion. We conclude from extensive performance evaluations that a relatively smooth channel estimation should be adopted for faster moving client while a more precise tracking is preferred in slower moving scenarios. More importantly, Prius is able to outperform exiting systems with near-optimal performance in terms of both QoE continuum and QoE fairness.

We would like to emphasize that Prius framework can not only be adopted in video on demand applications, but also in virtually any other HTTP-segment-based video delivery applications, e.g., video sharing in social networks and multi-camera streaming. This edge cloud assisted framework is indeed a promising paradigm to enhance tomorrow's video-rich mobile services. Future work shall be focused on joint rate adaptation and resource allocation by incorporating downlink scheduling into current framework. Such cross-layer network-side assistance is expected to achieve even further performance enhancement for future cloud-based mobile video delivery.

References

[1] European Telecommunications Standards Institute (ETSI), "Mobile edge computing - introductory technical white paper," https://goo.gl/HZyfPb.

[2] ETSI, "Mobile edge computing standard portal," https://goo.gl/0CquZr.

[3] Cisco White Paper, "Cisco visual networking index: Global mobile data traffic forecast update, 2014-2019," http://goo.gl/EqYUn9.

[4] I. Sodagar, "The mpeg-dash standard for multimedia streaming over the internet," *IEEE Multimedia*, vol. 18, pp. 62–67, Apr. 2011.

[5] C. Müller, S. Lederer, and C. Timmerer, "An evaluation of dynamic adaptive streaming over http in vehicular environments," in *ACM 4th Workshop on Mobile Video (MoVid)*, Feb. 2012, pp. 37–42.

[6] C. Liu, I. Bouazizi, and M. Gabbouj, "Rate adaptation for adaptive http streaming," in *ACM conference on Multimedia systems (MMSys)*, Chapel Hill, USA, Feb. 2012, pp. 169–174.

[7] S. Akhshabi, L. Anantakrishnan, A. C. Begen, and C. Dovrolis, "What happens when http adaptive streaming players compete for bandwidth?" in *ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, Jun. 2012, pp. 9–14.

[8] Z. Li, X. Zhu, J. Gahm, R. Pan, H. Hu, A. Begen, and D. Oran, "Probe and adapt: rate adaptation for HTTP video streaming at scale," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 4, pp. 719–733, Apr. 2014.

[9] V. Aggarwal, R. Jana, J. Pang, K. Ramakrishnan, and N. Shankaranarayanan, "Characterizing fairness for 3g wireless networks," in *Proc. of 18th IEEE Workshop on Local Metropolitan Area Networks (LANMAN)*, Chapel Hill, USA, Oct. 2011, pp. 1–6.

[10] S. Xiang, L. Cai, and J. Pan, "Adaptive scalable video streaming in wireless networks," in *ACM conference on Multimedia systems (MMSys)*, Chapel Hill, USA, Feb. 2012, pp. 167–172.

[11] J. Jiang, V. Sekar, and H. Zhang, "Improving fairness, efficiency, and stability in http-based adaptive video streaming with festive," in *ACM International Conference on Emerging Networking Experiments and Technologies (CoNEXT)*, Nice, France, Sep. 2012, pp. 97–108.

[12] S. Akhshabi, L. Anantakrishnan, C. Dovrolis, and A. Begen, "Server-based traffic shaping for stabilizing oscillating adaptive streaming players," in *ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, Feb. 2013, pp. 19–24.

[13] R. Houdaille and S. Gouache, "Shaping HTTP adaptive streams for a better user experience," in *ACM conference on Multimedia systems (MMSys)*, Chapel Hill, USA, Feb. 2012, pp. 1–9.

[14] L. De Cicco, S. Mascolo, and V. Palmisano, "Feedback control for adaptive live video streaming," in *Proc. of ACM conference on Multimedia systems (MMSys)*, San Jose, USA, Feb. 2011, pp. 145–156.

[15] J. Chen, R. Mahindra, M. A. Khojastepour, S. Rangarajan, and M. Chiang, "A scheduling framework for adaptive video delivery over cellular networks," in *ACM MobiCom*, Miami, USA, Sep. 2013, pp. 389–400.

[16] M. Zhao, X. Gong, J. Liang, W. Wang, X. Que, and S. Cheng, "Qoe-driven cross-layer optimization for wireless dynamic adaptive streaming of scalable videos over http," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, pp. 451–465, Mar. 2015.

[17] V. Joseph and G. de Veciana, "NOVA: Qoe-driven optimization of dash-based video delivery in networks," in *IEEE INFOCOM*, Toronto, Canada, Apr. 2014, pp. 82–90.

[18] D. De Vleeschauwer, H. Viswanathan, A. Beck, S. Benno, G. Li, and R. Miller, "Optimization of HTTP adaptive streaming over mobile cellular networks," in *IEEE INFOCOM*, Apr. 2013, pp. 898–997.

[19] A. Essaili, D. Schroeder, E. Steinbach, D. Staehle, and M. Shehada, "Qoe-based traffic and resource management for adaptive http video delivery in lte," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, pp. 988–1001, Jun. 2015.

[20] Z. Yan, J. Xue, and C. W. Chen, "QoE continuum driven HTTP adaptive streaming over multi-client wireless networks," in *IEEE International Conference on Multimedia and Expo (ICME)*, Chengdu, China, Jul. 2014, pp. 1–6.

[21] R. Mok, X. Luo, E. Chan, and R. Chang, "Qdash: a qoe-aware dash system," in *Proc. of the 3rd annual ACM conference on Multimedia systems (MMSys)*, Chapel Hill, USA, Feb. 2012, pp. 11–22.

[22] Z. Yan, Q. Liu, T. Zhang, and C. W. Chen, "Exploring QoE for power efficiency: a field study on mobile videos with LCD displays," in *23rd ACM International Conference on Multimedia (MM)*, Brisbane, Australia, Oct. 2015, pp. 431–440.

[23] A. Khan, L. Sun, and E. Ifeachor, "Qoe prediction model and its application in video quality adaptation over umts networks," *IEEE Trans. Multimedia*, vol. 14, pp. 431–442, Apr. 2012.

[24] D. Rodriguez, J. Abrahao, D. Begazo, R. Rosa, and G. Bressan, "Quality metric to assess video streaming service over tcp considering temporal location of pauses," *IEEE Trans. Consum. Electron.*, vol. 58, pp. 985–992, Aug. 2012.

[25] C. Alberti, D. Renzi, C. Timmerer, C. Mueller, S. Lederer, S. Battista, and M. Mattavelli, "Automated qoe evaluation of dynamic adaptive streaming over http," in *International Workshop on Quality of Multimedia Experience (QoMEX)*, Jul. 2013, pp. 58–63.

[26] X. Xie, X. Zhang, S. Kumar, and L. E. Li, "pistream: Physical layer informed adaptive video streaming over lte," in *ACM International Conference on Mobile Computing & Networking (MobiCom)*, Paris, France, Sep. 2015, pp. 413–425.

[27] A. M. Mutairi and U. A. Baroudi, "Ns-2 enhancements for detailed hsdpa simulations," in *Proc. of the 3rd International Conference on Performance Evaluation Methodologies and Tools*, Brussels, Belgium, Oct. 2008.

[28] 3GPP, "3GPP TS 25.214 v7.1.0 physical layer procedures(fdd)," 2006.

[29] *Progressive download and dynamic adaptive streaming over http*, Std. 3GPP TS 26.247 V12.1.0, 2013, Available: http://goo.gl/4EJbvd.

[30] J. Xue, D.-Q. Zhang, H. Yu, and C. W. Chen, "Assessing quality of experience for adaptive HTTP video streaming," in *IEEE ICME workshop on emerging multimedia systems and applications*, Chengdu, China, Jul. 2014, pp. 1–6.

[31] A. D. Baddeley, *Essentials of human memory*. Psychology Press, 1999.

[32] ITU, "ITU-R BT.500-13 methodology for the subjective assessment of the quality of television pictures," 2012.

[33] R. G. Brown, *Smoothing, forecasting and prediction of discrete time series*. Courier Corporation, 2004.

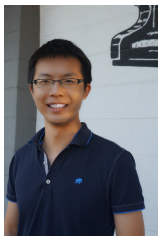
[34] ITU, "ITU-R M.1225 guidelines for evaluation of radio transmission technologies for imt-2000," 1997.

[35] A. Essaili, D. Schroeder, D. Staehle, M. Shehada, W. Kellerer, and E. Steinbach, "Quality-of-experience driven adaptive http media delivery," in *IEEE International Conference on Communications (ICC)*, Budapest, Hungary, Jun. 2013, pp. 2480–2485.

[36] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang, "Developing a predictive model of quality of experience for internet video," in *ACM SIGCOMM*, Aug. 2013, pp. 339–350.

[37] A. Giladi and T. Stockhammer, "Descriptions of core experiments on DASH amendment," MPEG N14619, Jul. 2014, Available: http://goo.gl/NJjnAz.

[38] Netflix to Pay Comcast for Smoother Streaming, Available: http://goo.gl/NPWxYm.



Zhisheng Yan is a Ph.D. student at Computer Science and Engineering Department, State University of New York at Buffalo. He received his B.S. and M.S. degrees from Shandong University and University of Science and Technology of China in 2010 and 2013, respectively. His research interests lie in the perception, processing and networking of multimedia content. Currently, his research is focused on mobile HTTP adaptive streaming and energy-saving mobile display.

Jingteng Xue received his B.S. and M.S. in EE from Southeast University in China in 2005 and 2009, and his Ph.D. from the State University of New York at Buffalo in 2014. He is currently a research engineer at Apple Inc. His focus includes mobile video coding, processing and networking.

Chang Wen Chen (F'14) received his BS from University of Science and Technology of China in 1983, MSEE from University of Southern California in 1986, and Ph.D. from University of Illinois at Urbana-Champaign in 1992. He is currently an Empire Innovation Professor of Computer Science and Engineering at the University at Buffalo, State University of New York. He was Allen Henry Endow Chair Professor at the Florida Institute of Technology from July 2003 to December 2007. He was on the faculty of Electrical and Computer Engineering at the University of Rochester from 1992 to 1996 and on the faculty of Electrical and Computer Engineering at the University of Missouri-Columbia from 1996 to 2003.

He has been the Editor-in-Chief for IEEE Trans. Multimedia since January 2014. He has also served as the Editor-in-Chief for IEEE Trans. Circuits and Systems for Video Technology from 2006 to 2009. He has been an Editor for several other major IEEE Transactions and Journals, including the Proceedings of IEEE, IEEE Journal of Selected Areas in Communications, and IEEE Journal on Emerging and Selected Topics in Circuits and Systems. He has served as Conference Chair for several major IEEE, ACM and SPIE conferences related to multimedia, video communications and signal processing. His research is supported by NSF, DARPA, Air Force, NASA, Whitaker Foundation, Microsoft, Intel, Kodak, Huawei, and Technicolor.

He and his students have received eight (8) Best Paper Awards or Best Student Paper Awards over the past two decades. He has also received several research and professional achievement awards, including the Sigma Xi Excellence in Graduate Research Mentoring Award in 2003, Alexander von Humboldt Research Award in 2010, and the State University of New York at Buffalo Exceptional Scholar - Sustained Achievement Award in 2012. He is an IEEE Fellow and an SPIE Fellow.